UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/915,751	07/26/2001	Peter Geiger	5143-02501	7482

7590 07/26/2004

Micheal P Adams
Winstead Sechrest & Minick P C
P O Box 50784
Dallas, TX 75250-0784

EXAMINER	
VERBRUGGE, KEVIN	

ART UNIT	PAPER NUMBER
2188	16

DATE MAILED: 07/26/2004

Please find below and/or attached an Office communication concerning this application or proceeding.

UNITED STATES PATENT AND TRADEMARK OFFICE

COMMISSIONER FOR PATENTS
UNITED STATES PATENT AND TRADEMARK OFFICE
P.O. BOX 1450
ALEXANDRIA, VA 22313-1450
www.uspto.gov

BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES

Application Number: 09/915,751
Filing Date: July 26, 2001
Appellant(s): GEIGER ET AL.

Ross Spencer Garsson, Reg. No. 38,150
For Appellant

EXAMINER'S ANSWER

This is in response to the appeal brief filed 6/1/04.

(1) Real Party in Interest

A statement identifying the real party in interest is contained in the brief.

(2) Related Appeals and Interferences

A statement identifying the related appeals and interferences which will directly affect or be directly affected by or have a bearing on the decision in the pending appeal is contained in the brief.

(3) Status of Claims

The statement of the status of the claims contained in the brief is correct.

(4) Status of Amendments After Final

No amendment after final has been filed.

(5) Summary of Invention

The summary of invention contained in the brief is correct.

(6) Issues

The appellant's statement of the issues in the brief is correct.

*(7) **Grouping of Claims***

Group 1

The rejection of claims 1-4, 22, 33-34, 36, 38-42, 55-61, 63, 67, 73, 78-79, 93-94, 97-98, 100, 102, 125, 129, and 131-134 stand or fall together because appellant's brief does not include a statement that this grouping of claims does not stand or fall together and reasons in support thereof. See 37 CFR 1.192(c)(7). In fact, Appellant states that these claims "form a first group" and the Examiner agrees. The representative claim for the group is claim 1.

Group 2

The rejection of claims 5-9, 17-21, 25-28, 35, 37, 43-45, 62, 64-66, 68-72, 74-76, 80-89, 95-96, 99, 101, 103-117, 126-128, and 130 stand or fall together because appellant's brief does not include a statement that this grouping of claims does not stand or fall together and reasons in support thereof. See 37 CFR 1.192(c)(7). In fact, Appellant states that these claims "form a second group" and the Examiner agrees.

However, while all the claims in the group are rejected under 35 USC 103, it is noted that the claims in the group have varying scope and can be grouped into subgroups by subject matter as done in the grounds of rejection.

(8) Claims Appealed

A substantially correct copy of appealed claim 66 appears on page 18 of the Appendix to the appellant's brief. The minor errors are as follows: claim 66 should depend on claim 65, not on claim 1765 (which does not exist).

(9) Prior Art of Record

5,699,539	GARBER et al.	12-1997

(10) Grounds of Rejection

The following ground(s) of rejection are applicable to the appealed claims:

Claim Rejections - 35 USC § 102

The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless –
>
> (b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States.

Claims 1-4, 22, 33, 34, 36, 38-42, 55-61, 63, 67, 73, 78, 79, 93, 94, 97, 98, 100, 102, 125, 129, and 131-134 are rejected under 35 U.S.C. 102(b) as being anticipated by U.S. Patent 5,699,539 to Garber et al., hereinafter simply Garber.

Regarding claims 1-4, 22, 33, 34, 38-42, 55-61, 93, 94, 98, 102, 129, and 131-134, Garber discloses a virtual memory management system and method using data compression.

Claim 1 is a representative claim for the group and is detailed here.

Garber's device is shown in Fig. 2 comprising CPU 102, virtual memory unit VMU 110, cache memory 108, primary memory 104, and secondary memory 106.

Garber's disclosure at column 1, line 40 through column 2, line 36 is particularly relevant to claim 1 and is pasted here:

> (6) Conventional Virtual Memory Management
>
> (7) The following is a brief explanation, for those readers not skilled in the art, of how conventional virtual memory management works. The virtual memory unit 110 enables programs executed by the CPU 102 to utilize an address space that is larger than the actual storage capacity of the primary memory 104. For instance, a computer with 8 MBytes (MegaBytes) of RAM might utilize a virtual address space of 16 MBytes, or even more. The VMU 110 uses a "page table" (discussed in more detail below) to store one "page table entry" for each virtual memory page in the computers virtual memory address space. One version of a page table is shown in FIG. 4 (although that page table contains features particular to the present invention).
>
> (8) Each page table entry includes an address value and status flags indicating whether the corresponding virtual memory page is stored in primary or secondary memory. Thus, the status flags in a page table entry indicate whether the address value in that entry corresponds to a primary memory location or a secondary memory location. For a computer system using pages that are 4 KBytes (KiloBytes) in size, a 16 MBytes virtual address space would occupy 4096 pages, and thus the page table would contain 4096 page table entries.
>
> (9) Since the CPU uses virtual addresses for its internal address computations, the virtual memory unit 110 translates each CPU specified virtual memory address values into a primary memory address value when the status flags in the corresponding page table entry indicates that the CPU specified address is currently stored in the primary memory 104. The translated address signal generated by the virtual memory unit 110 is then used to address the primary memory 104. When the CPU specified virtual memory address value is not currently stored in the primary memory 104, the page table entry accessed by

> the virtual memory unit will have a status flag indicating that fact, which causes the virtual memory unit to generate a "fault signal". The fault signal indicates the virtual page in which the fault occurred and also indicates, either directly or indirectly, the values stored in the address and flag fields of the corresponding page table entry.
>
> (10) Whenever a virtual memory fault occurs, a software procedure, typically called a virtual memory manager, locates a page of primary memory that can be mapped into the virtual memory page in which the fault occurs. If the virtual page upon which the fault occurred was previously swapped out to secondary memory, the contents of the virtual memory page are copied into the page of primary memory. The virtual memory manager then updates the page table entry for the virtual page to indicate the physical primary memory page in which the virtual memory page is now located, and then signals the virtual memory unit 110 to resume operation. Upon resuming operation, the virtual memory unit 110 again accesses the page table entry for the CPU specified virtual memory address, and this time successfully translates the virtual memory address into a primary memory address.
>
> (11) While there are many other aspects of virtual memory management not discussed here, the salient point here is that in standard virtual memory management subsystems, at any one point in time a virtual memory page is either in primary memory or secondary memory, or possibly in transition from one to the other.

From these passages it is clear that Garber's device receives a system memory access, as claimed, in the form of a virtual address from the CPU. His VMU 110 locates a page translation entry (what he calls a "page table entry" in paragraph 7 above and shows in Fig. 5) in a page translation table (what he calls a "page table" in paragraph 7 above and a "page map" at column 7, line 2 and in Fig. 4).

Once the page table entry for the desired page has been located in the page table, it is examined to determine whether the page is stored in primary memory or secondary memory. These operations are the well-known steps of virtual memory systems, swapping pages (chunks of data which are typically on the order of 4KB in size) back and forth from disk (also called secondary memory) to RAM (also called

primary memory, system memory, physical memory, or main memory). Since a computer cannot read data quickly from disk, the most recently used pages are stored in RAM, where a computer can access them more quickly. Since any page from the disk can be stored in any "page frame" (another name for a page location) in RAM, a page table is generally used to keep track of which pages from disk are currently stored in RAM and where they are stored. This is typically accomplished by creating a page table having a single entry for each page, each entry storing for its associated page the virtual page number (where it is on the disk), a physical page number (where it is in RAM) if it is currently in RAM, and various control bits, access rights, etc.

Garber's device builds on this well-known virtual memory technique of swapping pages in and out of RAM from the disk by compressing some of the pages in RAM. He explains the basis for his invention at column 2, line 37 through column 3, line 63. Essentially, he realized that compressing pages in RAM when they are not needed and decompressing them when they are needed would be quicker than loading pages from the disk. He found that pages in RAM can generally be compressed by a factor of 3, so a compressed page takes up 1/3 as much space as an uncompressed page. Therefore, in the simplest example, a prior art device (such as that shown in Garber's Fig. 1) might only have room for 2 pages in RAM while Garber's device (shown in Fig. 2), having the same size RAM, would be able to store 4 pages, 1 uncompressed, 3 compressed. This virtual "doubling" of the RAM size is what gives rise to the name of the commercial software product protected by Garber's patent: RAM Doubler ™.

Garber's device then is essentially a virtual memory system for managing the compression and decompression of pages within RAM and the swapping of pages back and forth between RAM and disk.

Since sometimes a page in Garber's device may be compressed, and since a compressed page is not readable, Garber's device must determine whether a desired page is compressed or uncompressed. If it is compressed, it must first be decompressed before it can be read from RAM.

The order of operations then is receive a virtual address, find the associated page table entry in the page table, determine if the page is stored in RAM (if not, load it from the disk), if the page is stored in RAM then determine if it is compressed. If the page is not compressed, it can be accessed immediately. If it is compressed, it must be decompressed first.

This process is spelled out at column 4, line 11 through column 5, line 8:

> (24) A virtual memory manager dynamically determines the number of pages of primary memory which need to be included in the work space, and moves pages of primary memory into the work space from the compression heap as needed. Virtual pages are selected to be swapped out of the work space to MappedOut storage on the basis of memory usage data. The virtual memory manager includes logic for preferentially compressing and storing swapped out virtual memory pages in the Compression Heap (in primary memory) so long as the MappedOut storage space includes a sufficient number of pages of primary memory to store those virtual memory pages, and for storing ones of the swapped out virtual memory pages in secondary memory when insufficient space is available in the Compression Heap. Thus, the size of the work space is variable and the proportion of swapped pages stored in primary memory and in secondary memory are also variable.
>
> (25) A page table stores a page table entry corresponding to each virtual memory page. Each page table entry specifies a tag value, corresponding to a location in the primary or secondary memory, and a plurality of status flags for indicating the status of the corresponding virtual memory page. The status flags includes a MappedIn flag having a True/False value that indicates when the tag value corresponds to a RAM page (in the active work space) in which the

corresponding virtual memory page is stored.

(26) A virtual memory unit (VMU) translates CPU specified virtual memory address values into primary physical address values when the MappedIn flag in the page table entries corresponding to the CPU specified virtual memory address values are True. The VMU generates fault signals when the MappedIn flag in the page table entries corresponding to the CPU specified virtual memory address values are False.

(27) The virtual memory manager responds to each fault signal from the VMU by updating the page table entry corresponding to the CPU specified virtual memory address value which caused the fault signal, so that the page table entry specifies a primary memory page and a MappedIn flag having a True value. The virtual memory manager includes a work space manager that defines a work space and a MappedOut storage space for storing virtual memory pages. The work space is located in a first portion of the primary memory. The MappedOut storage space is located in a second portion of the primary memory pages and in a portion of the secondary memory.

(28) A memory usage monitor stores memory usage data. The virtual memory manager includes swap out selection logic that selects, on the basis of the memory usage data, virtual memory pages to be swapped out from the work space to the MappedOut storage space. A MappedOut storage manager receives from the virtual memory manager the virtual memory pages selected to be swapped out, stores the received virtual memory pages in the MappedOut storage space, and adds the primary memory pages in which the received virtual memory pages were stored to the MappedOut storage space.

(29) The MappedOut storage manager includes a data compressor that compresses at least some of the received virtual memory pages prior to their storage in the MappedOut storage space, and a data decompressor that decompresses the compressed virtual memory pages when the virtual memory manager responds to fault signals caused by VMU faults on the compressed virtual memory pages.

Regarding claims 36, 63, and 100, Garber's system includes program instructions for an operating system that is not aware of the increased effective size of the memory as claimed.

Regarding claims 73, 78, 79, and 125, Garber's device includes the claimed compression/decompression engine.

Regarding claim 67, Garber shows the claimed page translation table in Fig. 4 as his page table or page map.

Regarding claim 97, Garber teaches that his device monitors the actual compression ratio achieved at column 20, lines 40-49 and claim 4.

Claim Rejections - 35 USC § 103

The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

Claims 5-9, 17-21, 25-28, 35, 37, 43-45, 62, 64-66, 68-72, 74-76, 80-89, 95, 96, 99, 101, 103-117, 126-128, and 130 are rejected under 35 U.S.C. 103(a) as being unpatentable over U.S. Patent 5,699,539 to Garber et al., hereinafter simply Garber.

Applicant has attempted to challenge the Examiner's taking of Official Notice. However, Applicant has not provided adequate information or argument that *on its face* creates a reasonable doubt regarding the circumstances justifying the Official Notice. See MPEP 2144.03 and In re Boon, 169 USPQ 231 (CCPA 1971).

Furthermore, as stated at MPEP 2144.03 C (emphasis added), "To adequately traverse such a finding, an applicant must specifically point out the supposed errors in the examiner's action, which would include stating why the noticed fact is not considered to be common knowledge or well-known in the art. See 37 CFR 1.111(b). See also Chevenard, 139 F.2d at 713, 60 USPQ at 241 ("[I]n the absence of any demand by appellant for the examiner to produce authority for his statement, we will not consider this contention."). A general allegation that the claims define a patentable invention without any reference to the examiner's assertion of official notice would be inadequate. If applicant adequately traverses the examiner's assertion of official notice, the examiner must provide documentary evidence in the next Office action if the rejection is to be maintained. See 37 CFR 1.104(c)(2). See also Zurko, 258 F.3d at 1386, 59 USPQ2d at 1697 ("[T]he Board [or examiner] must point to some concrete evidence in the record in support of these findings" to satisfy the substantial evidence test). If the examiner is relying on personal knowledge to support the finding of what is known in the art, the examiner must provide an affidavit or declaration setting forth specific factual statements and explanation to support the finding. See 37 CFR 1.104(d)(2). If applicant does not traverse the examiner's assertion of official notice or applicant's traverse is not adequate, the examiner should clearly indicate in the next Office action that the common knowledge or well-known in the art statement is taken to be admitted prior art because applicant either failed to traverse the examiner's assertion of official notice or that the traverse was inadequate. If the traverse was inadequate, the examiner should include an explanation as to why it was inadequate."

In this case, the Appellant's traverse was inadequate because Appellant has merely stated that "the Examiner has not met his burden of factually supporting his position that the claim elements not taught by Garber would have been 'obvious' or 'well-known.'" Appellant has not specifically pointed out the supposed errors in the Examiner's action, including stating why the noticed facts are not considered to be well-known.

In an effort to further clarify the Examiner's position, the Examiner has further explained the grounds of rejection below, clearly detailing the reasoning behind the rejection.

Regarding claims 35, 37, 62, 64, 99, and 101, in Garber's device, the operating system is apparently unaware of the increased effective size of the memory. However, it would have been obvious to one of ordinary skill in the art at the time the invention was made to make the operating system aware of the increased effective size of the system memory to provide enhanced memory control to the operating system. Since the operating system generally controls the allocation and usage of memory, it would be advantageous to the operating system to be aware of the increased effective size of the memory made possible by Garber's compression/decompression techniques.

Regarding claims 5-9, 25-28, 43-45, 74-76, 126, and 127, Garber does not disclose the claimed DMA channels but his device operates equivalently by transferring and receiving data to and from the VMU as needed. It would have been obvious to one

of ordinary skill in the art at the time the invention was made to use DMA since it provides an efficient data transfer, freeing up the processor to do other things. DMA (Direct Memory Access) was a well-known technique of transferring data from one location in memory to another location in the same or a different memory. It typically involved the CPU only at the beginning stage of the data transfer, allowing the CPU to do other things once the data transfer has begun. The CPU typically sent the starting address where the data to be transferred is located, the number of blocks to be transferred, and the location to where the data is to be transferred. Given this information, the DMA controller or device can complete the data transfer on its own with no further information from the CPU, allowing the CPU to perform other tasks while the DMA controller transfers the data. Once the data transfer is complete, the DMA controller typically informed the CPU of the completion.

Regarding claims 17-19 and 68-70, Garber does not disclose a page translation cache, but it would have been obvious to one of ordinary skill in the art at the time the invention was made to include such a cache (also known as a translation lookaside buffer or TLB) since TLBs were well-known in the art at the time and improved operation speed by caching recently used page address translations. This saves time by avoiding accessing the page table since if a translation is located in the TLB, then page table access is unnecessary. Only when the TLB does not contain the desired translation is access to the page table required.

The steps of looking in the TLB for a translation, and if not found, of obtaining it from the page table and storing it in the TLB were well-known at the time of the invention and would obviously be implemented if a TLB were implemented.

After the first Office action, Appellant challenged the Examiner's holding that TLBs were well-known. In the final rejection, the Examiner considered providing a reference but in fact the Applicants themselves have provided a reference on their latest 1449 which includes a TLB. Reference A5 (WO 97/23828) which was published well in advance of Applicants' filing date clearly shows a TLB in the cover figure which is Fig. 3. The reference extensively describes TLB operation at page 2, lines 24-33 and mentions the TLB in several other places. So Applicants' own submission is relied on as documented evidence that TLBs were well-known in the art at the time of the invention.

Regarding claims 65 and 103, Garber does not disclose the claimed scatter/gather DMA channels but his device operates equivalently by transferring and receiving data to and from the VMU as needed. It would have been obvious to one of ordinary skill in the art at the time the invention was made to use DMA since it provides an efficient data transfer, freeing up the processor to do other things. DMA (Direct Memory Access) was a well-known technique of transferring data from one location in memory to another location in the same or a different memory. It typically involved the CPU only at the beginning stage of the data transfer, allowing the CPU to do other things once the data transfer has begun. The CPU typically sent the starting address where the data to be transferred is located, the number of blocks to be transferred, and

the location to where the data is to be transferred. Given this information, the DMA controller or device can complete the data transfer on its own with no further information from the CPU, allowing the CPU to perform other tasks while the DMA controller transfers the data. Once the data transfer is complete, the DMA controller typically informed the CPU of the completion.

Garber does not disclose a page translation cache, but it would have been obvious to one of ordinary skill in the art at the time the invention was made to include such a cache (also known as a translation lookaside buffer or TLB) since TLBs were well-known in the art at the time and improved operation speed by caching recently used page address translations. This saves time by avoiding accessing the page table since if a translation is located in the TLB, then page table access is unnecessary. Only when the TLB does not contain the desired translation is access to the page table required.

The steps of looking in the TLB for a translation, and if not found, of obtaining it from the page table and storing it in the TLB were well-known at the time of the invention and would obviously be implemented if a TLB were implemented.

After the first Office action, Appellant challenged the Examiner's holding that TLBs were well-known. In the final rejection, the Examiner considered providing a reference but in fact the Applicants themselves have provided a reference on their latest 1449 which includes a TLB. Reference A5 (WO 97/23828) which was published well in advance of Applicants' filing date clearly shows a TLB in the cover figure which is Fig. 3. The reference extensively describes TLB operation at page 2, lines 24-33 and mentions

the TLB in several other places. So Applicants' own submission is relied on as documented evidence that TLBs were well-known in the art at the time of the invention.

Regarding claims 66 and 104, Garber's device includes the claimed compression/decompression engine since it performs compression and decompression.

Regarding claims 20, 21, 71, 72, and 105, TLBs were commonly fully-associative for operating speed reasons, typically because of the flexibility in placing an entry in any slot and the speed from being able to compare all entries at once.

TLBs are essentially caches that store page translations. To speed up the page translation process, it was known to include a page translation cache (TLB) to store the most recently used page translations so they can be provided immediately if desired. The size of TLBs was limited to typically on the order of 10 entries, for space reasons. This small size makes hit rate considerations very important and since fully associative TLBs have the highest hit rate, it would have been obvious to one of ordinary skill in the art at the time the invention was made to make the TLB fully associative.

TLBs (and all caches) are generally one of three types: fully associative, set-associative, or direct-mapped. Fully associative caches allow any translation to be placed in any slot, yielding the highest hit rate. Set-associative caches allow a single translation to occupy one of a few slots (equal to the number of sets used in the particular design), yielding a lower hit rate. Direct-mapped caches allow a single translation to occupy only a single slot, yielding the lowest hit rate of the three types. So

it would have been obvious to one of ordinary skill in the art at the time the invention was made to make the TLB fully associative to achieve the highest hit rate, and in fact, the most common TLB is the fully associative version for that very reason.

Regarding claims 80-89 and 106-117, the locations of the various elements are a matter of design choice, with certain advantages gained by placing certain elements in certain locations.

Placing the compression/decompression engine in the memory management unit or in a processor or on a memory module are all obvious places for various reasons. The memory management unit is in control of the swapping procedures and placing the engine there allows the most direct control of it. The processor has the most computation circuitry already so placing the engine there may allow the cheapest operation. The memory module contains the actual data to be compressed and decompressed so placing the engine there may provide the fastest operation.

Similarly, the placement of the memory management unit (CMMU) is a matter of design choice. Placing it in the system memory controller or coupled to the system memory controller is a decision the designer must make based on system considerations like complexity, chip placement, speed, etc.

Regarding is variable claim 95, Garber does not indicate that his page size is programmable, but the page size is a matter of design choice and therefore it would have been obvious to one of ordinary skill in the art at the time the invention was made

to make the page size programmable, allowing a system to use different size pages at different times. Smaller page sizes allow finer control but require more page addresses and therefore larger page tables and TLBs. Larger page sizes require fewer page addresses and therefore smaller page tables and TLBs, but also require more coarse control, since a greater amount of data exists on each page. Different applications have different optimal page sizes, so it may be advantageous to allow the user to program different page sizes at different times, depending on the applications being run at the time.

Regarding claim 96, Garber does not indicate that his maximum compression ratio is programmable, but it would have been obvious to one of ordinary skill in the art at the time the invention was made to make it programmable to affect various degrees of compression.

Regarding claim 128, Garber's device maintains the claimed lists as shown in Fig. 13 for example.

Regarding claim 130, Garber does not teach that his threshold ratio is programmable, but it would have been obvious to one of ordinary skill in the art at the time the invention was made to make it programmable to achieve a specific ratio. In Garber's device, the ratio is roughly 3:1, compressed pages to uncompressed pages.

(11) Response to Argument

Applicants argue that Garber's system "relies" on secondary memory (disk) to store compressed pages when insufficient primary memory is available in the compression heap in primary memory. The Examiner agrees. When necessary, Garber's device swaps compressed pages out to disk. But this is not always necessary, as indicated by Garber himself when he teaches that his device swaps "few, if any, pages out to secondary memory" (column 4, lines 2-3). He specifically mentions "if any" which is clear teaching that sometimes it is not necessary to swap any pages out to disk. At page 3, lines 19-20, he teaches that his device can provide twice the amount of primary memory "without having to swap any pages out to disk." He teaches at column 3, lines 31-32, that "Actually, in most cases, no pages will be swapped out to disk." And he teaches at column 3, lines 49-52 that "Use of the present invention allows the use of virtual memory while eliminating or substantially reducing the power usage associated with swapping pages to and from hard disk."

It is clear that when the size of the virtual address space being used is sufficiently small (when the number of pages being used is sufficiently small), all of the desired pages will be able to fit in his primary memory, avoiding all swaps out to disk.

In the Appellants' own specification, they teach that "pages from the compressed cache 240, which are maintained in a compressed format, can be moved to disk" (page 19, lines 11-15).

In any case, the claimed invention is anticipated by a device that swaps pages out to disk because the claims do not preclude swapping pages out to disk.

In a first interpretation of the claims, physical memory and system memory can include a disk. The broadest reasonable interpretation of physical memory includes a disk because a disk is a physical device. The broadest reasonable interpretation of system memory includes a disk because a disk is used by and is available to a system and is therefore a system device. Nothing in the specification clearly teaches that physical memory and system memory cannot include a disk.

In a second interpretation of the claims, where physical memory and system memory cannot include a disk, the broadest reasonable interpretation of the claims still allows them to be anticipated by Garber because the claims do not preclude a device that uses a disk in conjunction with physical memory. The claims require compression and decompression of pages in physical memory which Garber's device performs. Nothing in the claims precludes the use of a disk in conjunction with the physical memory.

Applicants argue that their "claimed invention does not rely on secondary memory", but neither is their claimed invention distinguished from a device that sometimes does use secondary memory, because a device which comprises primary and secondary memory (Garber's) also comprises primary memory, as claimed by Applicants.

Applicants further argue that in their claimed memory system, "the effective size of system memory is increased by storing the least used pages in a compressed format only in system memory" and they cite their specification at page 19, lines 9-11. But in the very next sentences in the specification, at page 19, lines 11-15, Applicants teach

that "In addition, pages from the compressed cache 240, which are maintained in compressed format, can be moved to disk or network in such format for future data storage, retrieval, or transmission over LANs or WANs. Thus, a second order benefit is achieved by storage of compressed pages in the I/O subsystem 300 instead of non-compressed pages" (emphasis added).

So not only do the claims not preclude use of a disk to store compressed pages, but the specification actually teaches the same.

Applicants challenged the Examiner's assertion that certain things were well-known at the time of the invention and demanded documented evidence to support the Examiner's assertion. In fact, the only specific thing that the Examiner asserted was well-known was the presence and use of translation lookaside buffers (TLBs) as page translation caches.

In an effort to support the Examiner's position that TLBs were well-known, the Examiner considered providing a reference but in fact the Applicants themselves have provided a reference on their latest 1449 which includes a TLB. Reference A5 (WO 97/23828) which was published well in advance of Applicants' filing date clearly shows a TLB in the cover figure which is Fig. 3. The reference extensively describes TLB operation at page 2, lines 24-33 and mentions the TLB in several other places. So Applicants' own submission is relied on as documented evidence that TLBs were well-known in the art at the time of the invention.

The Appellant's traverse of the Examiner's assertion that TLBs were well-known was inadequate because Appellant has merely stated that "the Examiner has not met

his burden of factually supporting his position that the claim elements not taught by Garber would have been 'obvious' or 'well-known.'" Appellant has not specifically pointed out the supposed errors in the Examiner's action, including stating why the noticed facts are not considered to be well-known. But in an effort to further clarify the Examiner's position, the Examiner has further explained the grounds of rejection above, clearly detailing the reasoning behind the rejection.

Similarly, regarding the other things that the Examiner had asserted were obvious, the Appellant's traverse was inadequate because Appellant has merely stated that "the Examiner has not met his burden of factually supporting his position that the claim elements not taught by Garber would have been 'obvious' or 'well-known.'" Appellant has not specifically pointed out the supposed errors in the Examiner's action, including stating why the noticed facts are not considered to be well-known. So in an effort to further clarify the Examiner's position, the Examiner has further explained the grounds of rejection above, clearly detailing the reasoning behind the rejections.

For the above reasons, it is believed that the rejections should be sustained.

Respectfully submitted,

Kevin Verbrugge
Primary Examiner
Art Unit 2188

***

July 21, 2004

Conferees

MANO PADMANABHAN
SUPERVISORY PATENT EXAMINER

Mano Padmanabhan
Supervisory Patent Examiner
Art Unit 2188

DONALD SPARKS
SUPERVISORY PATENT EXAMINER

Don Sparks
Supervisory Patent Examiner
Art Unit 2187

***

Jeffrey C. Hood
Conley, Rose, & Tayon, P.C.
P.O. Box 398
Austin, TX 78767